Title
Management Policy for Greater Computer Benefits: Friendly Software, Computer Literacy, or Formal Training

Journal
Social Science Computer Review, 12(3)

ISSN
0894-4393

Authors
Northrop, A
Kraemer, KL
Dunkle, DE
et al.

Publication Date
1994

DOI
10.1177/089443939401200304

Management Policy for Greater Computer Benefits: Friendly Software, Computer Literacy, or Formal Training

Alana Northrop, Kenneth L. Kraemer, Debora E. Dunkle, and John Leslie King

Using data from over 3,000 public employees in 46 U.S. cities in 1988, this article investigates three classes of factors commonly thought to affect computer use: training, friendliness of software, and user computer background. Computer use is analyzed as 11 specific tasks (such as programming, record searching) and is further broken down by organizational role of user, for example, manager and street-level employee. Some findings are that (1) the computer literacy or prior coursework of employees is more important to their computer use than how many years an employee has used computers; and (2) for most employees the user friendliness of programs is relevant, and weakly so, only for generic tasks such as searching a file or entering data. More generally, the data lead us to highlight training because it can be used to compensate for weaknesses in present software as well as in the computer literacy and experience of users. *Keywords:* computing benefits, friendly software, computer literacy, training.

An early theme of research on computing was on factors that lead to innovation adoption. Another strong theme was innovation outcomes—did computerization result in benefits. A third theme, and the one this research addresses, is use, what factors influence employee use of computers. Obviously benefits cannot be realized unless computers are utilized. It is common for executives and managers to complain about the lack of use of computers. A refrain heard by many who have done fieldwork on computing is "I'm not going to spend another dime for computers until you can show me that the computers we already have are being used." Moreover, some studies have even shown that many users were aware of only about 20% of the software's capabilities and used only about 10% (Kling & Jewett, 1991). Thus, use is an important third theme for research on computing (Leonard-Barton & Deschamps, 1988; Weiss & Birnbaum, 1989).

Social Science Computer Review 12:3, Fall 1994. CCC 0894-4393/94/$1.50.

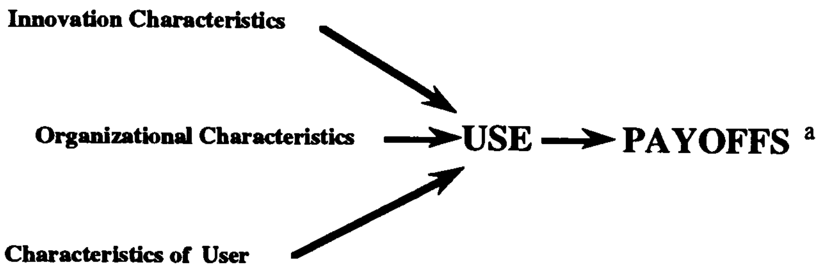

Figure 1 General model of factors influencing use and payoffs of an innovation. [a]Payoffs are not measured in this article but are in a previous article (Northrop, Kraemer, Dunkle, King, 1990).

Our interest is in the correlates of use generally, whether tied to the characteristics of the innovation, such as ease of use and software capabilities, or to the characteristics of the innovation users, such as computer experience and knowledge (Davis, Bagozzi, & Warshaw, 1989). The characteristics of the organization also need to be considered to the extent they influence the innovation and user. This general model of innovation dynamics has become a major focus of the recent literature on the economics of technological development and change (Rosenberg, 1982; Dosi, Freeman, Nelson, Silverberg, & Socte, 1988) (see Figure 1).

Applying this general model of innovation dynamics to computer use results in our focusing on software, users, and training characteristics specifically. To begin with, certain basic factors must be recognized. For example, the industry offers certain products that an organization has purchased. These products have particular characteristics that affect use, such as user friendliness. An organization also has a set group of employees on the job; these employees have a set amount of computer knowledge and experience. Finally, an organization has some type of training program to interface the first two. Together these factors form a system. This study explores how important each of the three factors are to computer use and whether they are equal in influence. It then draws implications for management policy (see Figure 2).

Hypotheses to be Tested

The specific hypotheses tested are the following: (1) the more user friendly the software, the more it will be used; (2) the more technically friendly the software, the more it will be used; (3) the more computer literate the employees, the more they will use the computer; (4) the more computer experienced the employees, the more they will use the computer; (5) the more formal the training in computer use, the more the use; and (6) the more professional the computer trainer, the more the use (see Figure 2).

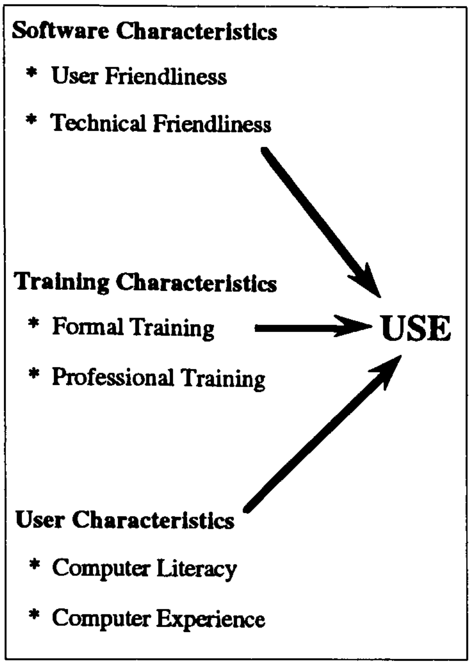

Figure 2 Model of factors explored that may influence computer use.

These are not new hypotheses. In fact, they have been studied before by others and are considered common assumptions, relied on in day-to-day operations. For example, many organizations consider how friendly a software package is before purchasing it. Moreover, job applicants are increasingly asked about their computer experience. But the above hypotheses have tended to be explored in case studies or industry specific studies, which have all been fairly small samples. Given that our sample is 3,000 employees in 46 cities, our data cut across organization and employee idiosyncrasies, allowing for wider testing of the hypotheses.

This article also uniquely explores the premise that the influences of friendliness of software, computer background of users, and training will vary depending on the task for which the employee is using the computer. Obviously, some tasks are more optional than others or vary in whether manual sources are also available. Moreover, tasks have different levels of complexity inherent in them; for example, one can just enter data or one can statistically analyze performance data for individual employees or units. The latter task appears to require more training or computer experience to accomplish. Thus, we expect support for the above hypotheses to vary by task, especially task complexity. The following tasks are analyzed: code or enter data, search for or retrieve records, process text, program the computer, prepare graphics, send mail, compute statistics using a statistical package, perform financial and/or budgetary calculations, perform engineering calculations, develop schedules for manpower and/or projects, and construct or update files.

Finally, the use of the computer to perform these 11 tasks is analyzed by one's role, whether managerial, staff/professional, or desktop and street-level employee. It is hypothesized that one's role influences the type of tasks performed (as job descriptions indicate) as well as certain user characteristics. Consequently, we expect some variations in which factors influence computer use depending on one's role in the organization. In sum, we expect some consistent trends in what factors influence use with some deviation by complexity of task and role in organization.

Methods and Data

The data for this article were collected as part of two studies referred to as URBIS I and URBIS II,[1] which were conducted through the Public Policy Research Organization at the University of California, Irvine, and supported by grants from the National Science Foundation. The sampling method here is derived from the URBIS I design, an overview of which is presented below. Detailed published discussions are contained in Kraemer, Dutton, and Northrop (1981) and Danziger and Kraemer (1991).

In 1976, information was gathered from 42 cities in the United States. These cities were chosen using an innovative design to sample purposively different types of cities rather than to sample randomly all cities per se. Through an initial mail survey inquiring about computer services in every city in the United States over 50,000 in population ($N = 403$), data were collected to stratify cities on six policy variables. These included the degree of (1) automation, (2) centralization, (3) data integration, (4) technical sophistication, (5) user involvement, and (6) charging for services. Thus the sample reflects leading-edge cities not simply in technology but also in computing practices. Each variable was dichotomized, yielding 64 possible combinations. Forty groups out of the possible 64 were randomly selected, and then a city in each group was randomly selected. Two additional cities were added because of resource allowance. In 1988, 5 of the original cities were unable to participate again; therefore, new cities were added to the original sample. Nine new cities were added and were chosen to reflect the single most important change in local government computing since the mid-70s—the proliferation of personal computers, especially among cities between 50,000 and 99,999 in population. Consequently, the 1988 sample represents leading-edge cities for that year and thereby permits us to address use adequately in 1988. The 46 cities in the 1988 sample reflect the diversity, although not proportionately, of U.S. cities in terms of population, region, and current state of computing practices.[2] The data for this article come from the 1988 sample.

The methods for this article focus on leading-edge cities and the computer users within those cities as predictors of where the majority of cities and government employees will be in the future. Thus, whereas the sample is biased toward more technically advanced cities and toward those city employees who use computers and information systems, the design has a distinct advantage for the purpose of this analysis. Specifically, the purpose is to study factors that affect computer use. Technically advanced cities would by definition be the most extensively automated, and therefore use could not only potentially vary the most but also would be the most relevant as an issue. This article is also concerned with giving policy advice, and thus the sampling frame uses sites that are more technologically advanced as predictors of the common state of computerization in the future.

An average of 100 respondents per city, 4,940 total, completed and returned anonymous questionnaires on computing use, roughly an 80% response rate for 1988.[3] The respondents represent policy makers, department managers, staff professionals (e.g., policy analysts, planners, budget and personnel analysts, engineers), street-level workers (e.g., welfare and public health workers, building inspectors), counter-top workers (e.g., collection clerks, dispatchers, building and planning and zoning "counter" clerks), and desk-top workers (e.g., clerical, bookkeepers). Those respondents who indicated that they never completed *any* of the 11 computer tasks were dropped from analysis because computer use was not an issue for them.[4] In addition, detectives and patrol officers are not analyzed in this article but are in a separate article because the nature of police computer searches is distinctive from all other computer uses both in criticalness of the factor (e.g., a patrol officer has a car pulled over and wants to search for outstanding warrants) and in open-endedness (e.g., detective searches for MOs, aliases, incorrectly spelled names, etc.). Moreover, police computer installations tend to be autonomous and have a unique history.[5] The resulting sample, minus police, used in this article is 3,143.

The six hypotheses discussed in the introduction are tested using individual-level measures because, obviously, how user friendly the software package is, for example, can be viewed only from the perspective of the individual. Also, years of computer experience and computer coursework—individual level measures—are two independent variables.

The data are based on the perceptions and/or factual recall of employees. In general it can be argued that perceptions or recall are not necessarily the most valid measures. Yet, usage of computer files is heavily dependent upon perceptions of whether one knows how to use them. In addition, perceptions provide the most direct measure possible for some factors, such as user friendliness of software. Thus, perceptions as the measure of key variables in this article may be the most valid measure possible. Moreover, recent research has shifted to exploring user perceptions of system effectiveness as a primary measure of information systems success (Srinivasar, 1985).

The measurement of all variables involved in the hypotheses is discussed in Table 1 notes. It is important to note that the measures allow for the imprecision of recall where appropriate. For example, the measure for use of the computer asked respondents to recall how often they perform 11 tasks with the following response choices: never, at least once a year, several times a year, a few times a month, a few times a week, and daily. Other measures are more precise where recall allowed it. For example, have you ever taken a course in programming, yes or no. The issue of the reliability and thus validity of measures based on recall certainly depends on whether the average respondent can recall the event with the level of precision demanded by the response choices. Furthermore, any measures that were built from more than one variable (user friendly and technically friendly) were based on a factor analysis with an acceptable standard high eigenvalue (.80 or higher).

Finally, Pearson's r is the statistic used to analyze the data. Type of employee is not an interval-level measure but is not included in any regression equation, instead it is hypothesized to be a specification test variable and is so analyzed. Five of the independent variables are at least interval in level of measurement; the sixth one is ordinal but is coded with the presumption of a zero value. The dependent variable, use, is ordinal but has a real zero and is coded as an interval measure, that is, 0, 1, 2, 3, 4, 5. The possible problem of a limited dependent variable is serious only in extreme cases, such as a large proportion of observations is in a small number of the categories given, which does not apply here (Fox, 1991). It is important to note that the correlations were run with an ordinal measure (gamma) and almost identical numbers were obtained. The decision to go with Pearson's r was based on the latter findings and because regression analysis could then be done with consistency in presentation.

Findings

Before the six hypotheses are tested, we need to acknowledge that payoffs are being realized from the use of computer systems. Other research based on the 1976 and the 1988 data found general payoffs across city departments in several areas (Northrop, Kraemer, Dunkle, & King, 1990; Kraemer et al., 1981; Kraemer & Northrop, 1989; Danziger & Kraemer, 1985; Northrop, Dutton, & Kraemer, 1982). Thus, we can probably rule out a possible major inhibitor to computer use, that is, its worthlessness. Now other factors that can affect use and thereby payoffs need to be addressed.

Software Characteristics (H_1 and H_2)

Any organization is inherently confined by the computer software systems already implemented, given that substantial costs have been

Table 1 Pearson correlations* between software, employee, and training factors and frequency of computer tasks[a]

	User friendly[b]	Tech friendly[c]	Computer literate[d]	Computer experience[e]	Formal training[f]	Professional training[g]
Search for data	.18**	.03	.03	.08**	.11**	.01
Enter data	.14**	.02	.04**	.06**	.10**	−.01
Update files	.09**	.05**	.13**	.09**	.06**	−.04**
Process text	.01	.13**	.14**	−.01	.10**	−.11**
Electronic mail	−.00	.08**	.10**	.09**	.10**	.01
Program	−.01	.06**	.27**	.05**	.04**	−.13**
Statistical analysis	−.01	.06**	.17**	.08**	.06**	−.09**
Financial calculations	−.06**	.08**	.20**	.16**	.05**	−.15**
Engineering calculations	.01	.06**	.14**	.03**	−.02	−.08**
Scheduling	−.01	.06**	.15**	.06**	−.01	−.10**
Graphics	−.04	.08**	.25**	.08**	−.01	−.20**

a = Frequency choices were "never" and "at least once a year," "several times a year" and "a few times a month," "a few times a week" and "daily." Respondents were dropped from analysis who never did *any* of the above tasks (please see Note 4). All calculations above .08 are underlined to indicate more than weak association.

b = User friendly: A mean value based on respondent's response to the following three questions about the computer application that they use the most in their work: hard to learn (1) to easy to learn (5); hard to use (1) to easy to use (5); reference manual hard to understand (1) to easy to understand (5). These three questions had an acceptable eigenvalue based on a factor analysis and were distinct factors from technical friendly.

c = Tech friendly: A mean value based on respondent's response to the following four questions about the computer application that they use the most in their work: lacks necessary capabilities (1) to has all capabilities I need (5); has lots of bugs (1) to bugs worked out (5); continually being changed (1) to fairly stable (5); needs to be changed and we can't get it changed (1) to needs to be changed and it's going to be taken care of (5). These four questions had an acceptable eigenvalue based on a factor analysis.

d = Computer literate: Range 0 to 2. A 2 represents having taken a course in programming *and* having participated in any courses, conferences, or seminars providing a general background regarding what computers can do and how they do it.

e = Computer experience: Number of years of direct involvement in using computers or computer-generated information. Range was 0 to 25 for 99% of the respondents. Half had less than 6 years.

f = Formal training: Formal training was recoded from total number of hours of training to 0 for none and 1 for 1 or more hours. Based on prior analysis, the issue was whether one had training or not. Interestingly, no other grouping or nongrouping of hours changed the correlations. The data revealed the distinction was between none and any.

g = Professional training: Categories were: self-trained including manual or computer-aided instruction software, trained by co-worker, trained by supervisor, trained by organization's or outside computer professional. The emphasis is on from whom one got the majority of training needed to begin using computing on the job.

*Pearson correlations were computed for several reasons. In brief: (1) same results as ordinal gamma measure; (2) allows for consistency in presentation with later regression results; and (3) the two variables have a true zero. See "Methods and Data" section for more detail. See Figure 3 for summary of partial correlations from regression equations.

** $p \leq .05$ Note that statistical significance is based on both the strength of a relationship and the size of the sample. Although our sample is large, its size is not sufficiently large enough to make all correlations significant. Thus, strength of association plays a role in these data's statistical significance.

Table 2 Frequency of performing computer tasks

	Seldom/Never[a] (%)	Occasionally[b] (%)	Frequently[c] (%)
Search for data	14	24	62
Enter data	31	21	47
Update files	45	26	29
Process text	37	22	40
Electronic mail	79	9	13
Program	80	13	7
Statistical analysis	78	18	5
Financial calculations	62	23	15
Engineering calculations	93	4	3
Scheduling	82	15	3
Graphics	80	17	4

a = Once a year or never.
b = Several times a year or a few times a month.
c = A few times a week or daily.

incurred. This section explores how constrained, if at all, computer use is by the characteristics of an organization's software. Great emphasis is placed during the purchasing decision process on user and technical friendliness of software packages. Is this emphasis as critical to computer use as thought?

H_1. The More User Friendly the Software Is, the More It Will Be Used
Use of the phrase user-friendly system (computer, software, or application) has become widespread in both the computer science and the MIS literature in recent years. This emphasis upon systems that are perceived by the user to be easy to understand and easy to use is considered to be very important if computer systems are to be widely used by laypersons (DeSanctis & Courtney, 1983; Davis et al., 1989; Newcomer & Caudle, 1991; Davis, 1989). User-friendly systems invite initial trial, experimentation, and exploration that will form a base of subsequent routine work. Our measure of user-friendliness is based on individual users' responses to three questions about whether the computer application that they use most in their work is hard or easy to learn and to use and whether the manual is hard or easy to understand (see Table 1b).

Surprisingly, user friendliness has a statistically significant and positive effect on the performance of only 3 tasks, which are the most common computer tasks: data entry, update, and search (see Table 1). Eighty-six percent (86%) of the employees search records, 68% enter data, 62% process text, and 55% update files. Fifth is financial calculations, 38% perform this task (see Table 2). What distinguishes entry, update, and search from processing text is that the former 3 tasks could be said to form a group that encompasses general (nondedicated) file use. In other words, one has to either enter, update, or search a file, but one does not have to process text, calculate statistics, or program. Given that entry, update, and search are among the most common across a range of computer tasks and are a generic group in terms of computer use, they will be performed by employees with the greatest variety in general skill level of any tasks. Thus, it makes sense that the more user friendly the program, the more the computer will be used to perform the most common and generally most basic computer tasks because helpfulness and ease of using the software are more important for those tasks given the range in skill and background of the users. Moreover, given that entry, update, and search form a generic group of tasks for which the computer is used, it is likely that many of these tasks have specifically been designed with user friendliness in mind. But there appears to still be room for improvement.

Interestingly, user friendliness has a significant but very weak negative effect on the likelihood of using the computer to perform financial calculations. In other words, the harder the application is to learn and use, the slightly more likely an employee calculates financial data. These findings are extremely weak. Perhaps the data are reflecting the frustrating experiences of employees in 1 or 2 cities who have terribly unfriendly finance packages but the employees have no choice but to use them often, given the dictates of their jobs. Our field work substantiates that such situations do exist. Moreover, other studies have also indicated this to be the case (Gasser, 1988).

In sum, user friendliness is not relevant to the performance of many tasks; and to the tasks where it is relevant (search, entry, update, financial calculation), user friendliness is found not to be a major determinant of frequency of computer use because the correlations tend to be low (Table 1). Thus, an organization is not a captive of the user friendliness of its programs.

H_2. The More Technically Friendly the Software, the More the Package Will Be Used

If a software package has a lot of bugs and/or is continually being changed to get rid of the bugs or to improve its capabilities, the result is likely to be frustration on the part of the user. What we call "technically unfriendly" systems are discouraging for users, interfering with and interrupting their work and creating additional work as mistakes must be corrected and workarounds developed and implemented. Thus, hardware and software developers have persistently sought to make their systems more technically friendly.

Our measure of technical friendliness is a factor based on individual users' responses to four questions about whether the computer application that they use most in their work has all the capabilities needed, has lots of bugs, is continually being changed, and needs to be changed (see Table 1c). Perceived usefulness, that is, having the capabilities needed, has been found to be a key factor to use and to be more important than ease of use or user friendliness (Davis et al., 1989; Davis 1989; Srinivasar, 1985).

This study's data suggest that the technical friendliness of the software is also important to overall use but is not key. There is significant but weak support for the hypothesis for all tasks but search and entering data, the two most commonly performed tasks (see Table 1).

One possible reason why technical friendliness is relevant for some tasks and not others may have to do with the sophistication of the tasks; with the more sophisticated tasks (nongeneric) requiring somewhat more technical ease if they are to be done more frequently, with the exception of updating a file. Updating a file is a generic task but can be more complicated than searching a file or entering data and thus benefits from technical friendliness, as do all other nongeneric tasks. It is of note that technical friendliness is most important for word processing.

In sum, user friendliness is a weak factor positively influencing the performance of the most common and generic tasks (search, entry, update), whereas technical friendliness is an even weaker factor influencing the performance of the more sophisticated tasks. In other words, to get employees to do the most common tasks on the computer, pay some attention to the user friendliness of programs available. To get employees to perform specific dedicated tasks more frequently, it is relevant to pay some attention to the technical friendliness of programs available. It should also be noted that there is great variability among systems in terms of both user and technical friendliness. Thus, there is much room for improvement in both areas.

User Characteristics (H_3 and H_4)

This section of the findings deals with individual characteristics that are likely to influence the degree to which an employee uses the computer. This topic is an important contextual factor to be considered. It is important to know whether an employee's computer background is irrelevant, a minor factor, or a major factor influencing computer use. In other words, how constrained is management by the degree of computer literacy and experience of its employees already on board? And, should these factors be taken into account in hiring decisions?

H_3. The More Computer Literate an Employee, the More Likely He or She Will Use the Computer

This hypothesis is based on research that shows that literacy results in computer confidence and therefore encourages computer use (Lucas, 1989; Shangraw, 1986). Computer literacy is measured here by a simple 3-point scale, ranging from 0 for "no programming skills" and "never participated in any courses, conferences, or seminars providing a general background regarding what computers can do and how they do it" to 2 for having both "programming skills from a course" *and* "prior course or seminar on what computers can do and how they work." A mark of 1 represents having "only a programming course" *or* "a seminar on computers in general." Among all users, 35% scored a 0, and 27% scored a 2.

The findings are quite consistent across the tasks examined in Table 1. That is, for all computer tasks except search, computer literacy has a significant, weak-to-low influence on use. These are our strongest correlations across the board. The lack of a relationship between literacy and the search task might have general explanations. The first is that although search might be a complex task for some employees, it involves simple fact retrieval for most people. The second explanation is that most people do some computerized searches. It was mentioned earlier that 86% of employees search records at least occasionally. Thus, it might be that people have so much experience with this task compared to any other task that literacy is not a relevant issue. In sum, coursework pays off in employees' turning to the computer to perform all tasks except search, the most common or universally performed task.

H_4. The More Computer Experienced an Employee, the More He or She Will Use the Computer

Years of past direct involvement in using a computer, computer software, or computer-generated information should build confidence in and knowledge of the computer, and thereby beget reliance on the computer (see Table 1).

The data indicate the hypothesis is significant for all tasks except processing text. Financial calculations have the largest but still low correlation (Table 1). Finance computer applications are one of the two oldest types of local government applications, police being the other. It may be that because most cities introduced computing first into police and finance that in these departments computer experience becomes a more relevant and complementary factor to computer literacy in determining use. Processing text might not be affected by employee computer experience because if you have to process text you just do it. For many employees, whether to use a typewriter or computer is no longer an issue, because only a computer is on their desks now.

In conclusion, the more computer literate the work force is, the more computers will be used to do all types of tasks except the most common, that is, searches. And a comparatively weaker influence is years of sheer experience with computers, which acts as only a slight encouragement to computer use but more so for financial calculations. Therefore, job applicants should be screened more on coursework than on years of computer experience where computer skills are relevant in a job.

Training (H_5 and H_6)

So far the findings suggest that software characteristics are relevant but weak determinants of frequency of use. More important influences to use are the computer backgrounds of employees. Coursework or literacy is the strongest determinant, although still low, with years of computer experience weaker. Both the software and employee characteristics are givens, i.e., there is little room for management policy to change them quickly, but training is much more open to change and can be the key intermediary between the software and employees on board.

H_5. The More Formal the In-House Training in Computer Use, the More the Use

This hypothesis is based on research that shows that any formal training should convey more information about how to use the computer in one's job than go-it-alone learning or informal co-worker advice and that knowledge begets use (Rivard & Huff, 1988; Sein, Bostrom, & Olfman, 1987; Cheney, Mann, & Amoroso, 1986; Zmud & Lind, 1985). The measure used was number of hours of formal training (see Table 2).

The hypothesis is slightly supported for all the tasks, except for graphics, engineering calculations, and scheduling (Table 1). The latter 3 tasks are principally done by engineers in the cities, and engineers have a very sophisticated and long professional training period prior to employment in contrast to most other city employees. Thus, it is not surprising that in-house formal training on how to use the computer in one's city job has no effect on engineers' use of computers because they probably had the training before they came on board. In sum, training results in a slight increase in overall use of the computer.

H_6. The More Professional and Superior the Computer Trainer, the More the Use

The assumption is that learning how to use the computer on one's own is harder than learning from co-workers who know the ropes. And, learning from a supervisor puts the imprimatur of authority on encouraging computer use along with the word of experience. Finally, it is assumed that computer professionals, whether inside or outside the organization, have the most knowledge of the software and thus would be capable of providing the most detailed training. Thus, the measure of professional training is a 4-point scale, ranging from self-taught to trained by a computer professional.

Without question, this sixth hypothesis is not supported. In fact, the more professional the training the significantly less likely an employee is to do all tasks except search, enter data, and electronic mail (see Table 1). These findings were curious. To explore the training issue further, the training variable was also analyzed as a series of dichotomies, that is, self-trained versus all other, trained by peer versus all other, and trained by professional versus all other. That analysis revealed that employees who learned the majority of what they needed to know to begin to use the computer in their jobs on their own were significantly more likely to perform all tasks (.08 to .31), except search and enter data. Conversely, employees who learned the majority of what they needed to know to begin to use the computer in their jobs from co-workers were significantly less likely to perform all tasks (−.09 to −.25), except search and enter data. And these findings held up when we did the analysis among the subgroups of professional and nonprofessional employees. Also, learning the majority of what one needed to know from a computer professional brought no increase or decrease in computer use.

On the surface, these results suggest that the best training practice is to hand an employee a computer and a book or videocassette. The results, however, bear further examination.

First, the above findings support self-learning, but the data also support formal training and coursework. Thirty percent of the respondents were initially self-taught, 37% were taught by co-workers, and 33% by computer professionals. This rather even distribution suggests that any of the training approaches are appropriate. We know that formal training, any amount, even one hour, makes a positive difference in computer use. Moreover, computer literacy, that is, coursework, is a consistent and, in fact, the strongest predictor investigated of frequent computer use.

Second, the self-trained group may reflect truly motivated employees, the kind who devote hours on their own to learning how to use the computer, hours beyond what peer teaching and classroom teaching demand. In other words, self-learners are unique because they are motivated and this motivation combined with the confidence in learning one task mostly on one's own allows employees to go on and learn other tasks, again on their own. But these same employees can benefit from formal training. In fact, their initial training if done by others probably was helpful but did not cover as much as some employees wanted to learn. This may be the key issue. Hands-on experience is invaluable, fiddling with the computer or just trying to do your work on your own is how one really learns to use the computer whether one is fascinated by computers or just needs to use them to do one's work.

In summary, it appears then that any formal training is useful to increasing computer use. In addition, given (1) that the professionalism-of-training question emphasized *initial* training for using the computer and (2) that formal training was significantly related to use, it may be that formal training *after* initial use of the computer is where emphasis should be placed. In other words, sometimes co-worker training is fine, and sometimes self-training is fine as the initial source of training for using the computer. But once an employee has some experience with using the computer, then formal training can expand use because specific questions, perhaps unique to an individual or too sophisticated to deal with at first, can be addressed. Moreover, initial training by others may not always be able to teach most of what is needed. Hands-on experience is invaluable and should be acknowledged in any initial training program. More advanced programs should be planned as necessary components to any initial training. Employees should be prepared psychologically to spend time on their own learning to use the computer, realizing that the time and frustration involved are natural.[6]

Relative Influence of Software, Characteristics, User Characteristics, and Training

The findings indicate that there is variation in the factors that influence employees to use computers for different tasks. Software characteristics appear to be particularly distinct in influence by task, with generic tasks requiring more user friendliness and more sophisticated tasks requiring more technical friendliness in applications (Table 1). In contrast, literacy, computer experience, and formal training have influence across tasks although they vary by degree of influence (Table 1).

The degree of independent influence of the three factors—friendliness of software, user background, and training—can be shown by regression analysis. The 11 regression equations, one for each task, show that user friendliness is the most important influence on the search and enter tasks (Figure 3). And computer literacy is the key variable influencing employee performance of the 8 sophisticated tasks (Figure 3). Any variable coming in second in the equations explained less than 2% of use. Variables coming in first explains 1% to 12% of use.

Types of Computer Tasks and Employee Role

But perhaps tasks are not the same for all employees. For example, when a department head does a search of a file, it may be a more voluntary and exploratory search than when a traffic-ticket counterperson does a search. If so, then the department head's search may be influenced by different factors from the clerk's. To explore this possibility, all 11 regressions were repeated for managers ($N = 799$), staff professionals ($N = 782$), and desktop, countertop, and street-level employees ($N = 1492$). Given that 33 separate regression equations resulted, the findings are only summarized in Figure 4. Only significant results are reported. In each case only one independent variable is discussed because no other independent variables in the regression equations added any explanatory value over 3%.

Computer literacy was the main influence for each of the three sets of employees in terms of their performance of programming,

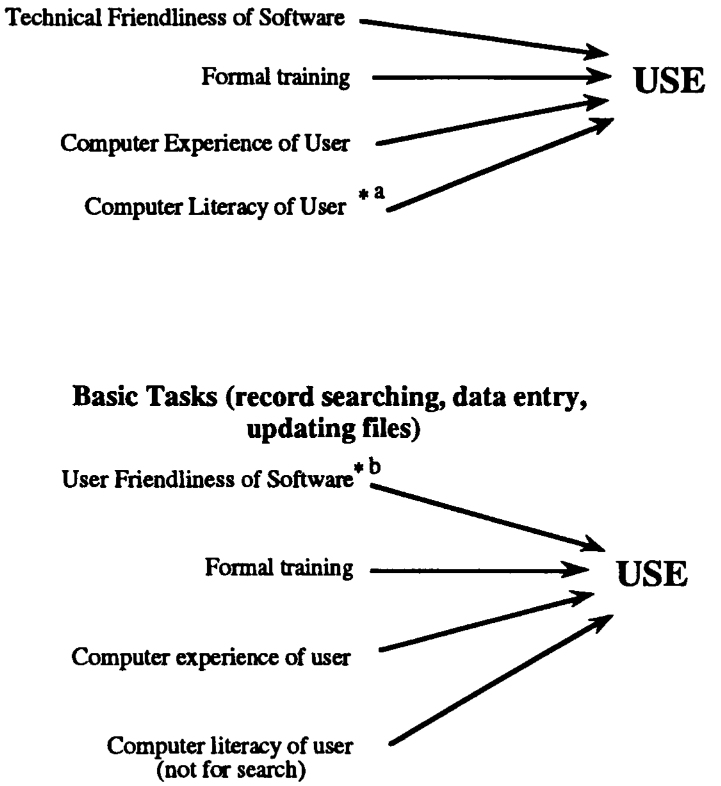

Figure 3 Significant factors influencing degree of computer use by task complexity. *Key independent variable based on regression equations. The starred variable consistently entered first. Any variable coming in second in the regression equations explains less than 2% of the variation of task use. [a]For all tasks except electronic mail where formal training is key variable. [b]For search and enter tasks. Literacy is the key variable for update task.

statistics, engineering calculations, scheduling, and graphics. Yet there were also differences. For searching a file and entering data, managers were most influenced by the user friendliness of programs whereas professionals and other employees were most influenced by their own computer experience. For updating a file, managers were again most influenced by the user friendliness of programs, but the other employees were most influenced by their computer literacy.

Word processing was most often done by initially self-taught managers and professionals and by the more computer literate desktop or street-level employees.

For nonmanagers, formal training most influenced their use of electronic mail, but for managers the key was the technical friendliness of the program.

Finally, the key factors influencing the performance of financial calculations for managers was their own self-training; for profession-

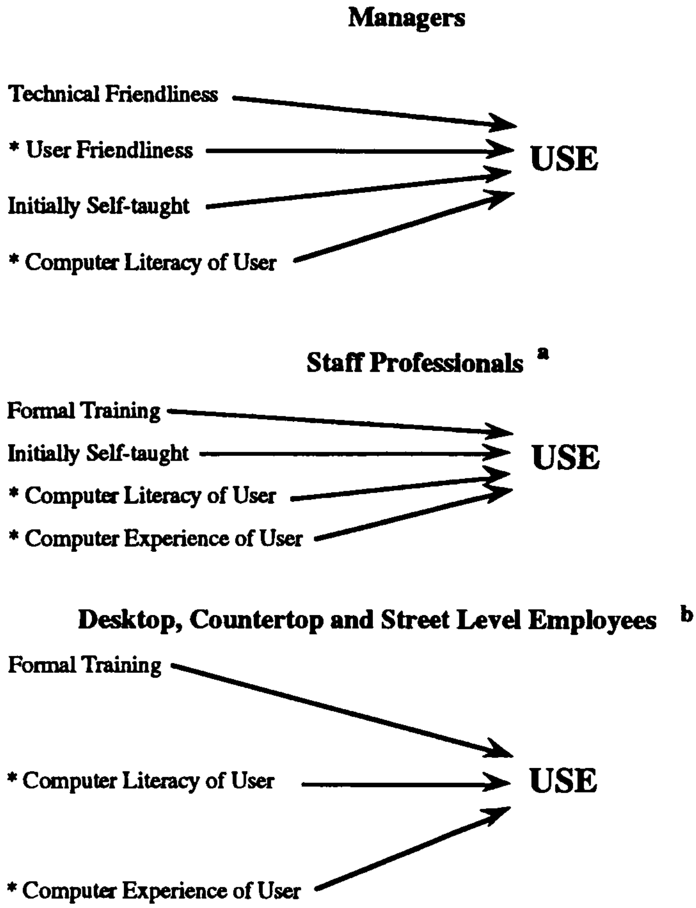

Figure 4 Significant primary factors influencing degree of use of any of 11 tasks. *Factors that had the most consistent effects across 11 tasks. [a]Policy analysts, planners, budget and personnel analysts, engineers. [b]Welfare and public health workers, building inspectors, collection clerks, dispatchers, building and planning and zoning counter clerks, clerical, bookkeepers.

als, it was their computer experience; and for the other employees, it was their computer literacy.

Looked at another way, for nonmanagers, computer literacy and computer experience are the key factors to the frequency of their performing computer tasks. For managers, computer literacy and user friendliness stand out as the key factors influencing their computer use (Figure 4). Another study argues that executives should be able to begin using the software without "*any* prior training with the system" (Mohan, Holstein, & Adams, 1990, p. 438). So user friendliness is particularly important for managers especially because two big reasons why they do not use the computer is that they lack computer training and experience and are unwilling to spend the time to gain the information (Mohan et al., 1990). Thus, user friendliness is a substitute for computer literacy among managers.

These variations by employee status are relevant to the prior findings, especially, we believe, in the importance that training should be in any organization concerned about computer use and computer payoffs.

Conclusions

Computer use is necessary for organizations to realize the payoffs from computerization. Thus this article sought to investigate how specific characteristics of software, users, and training influence computer use. It was thought that knowledge of those relationships would assist public managers who seek to encourage greater computer use among government employees in order to achieve greater payoffs.

Other research suggests that local government employees are very pro-computer and overwhelmingly think computers improve the way their jobs are done (Kraemer & Northrop, 1989). So one no longer has to sell computers to employees to overcome their fears and resistance. But how constrained is employee computer use by the characteristics of the software already purchased, by their own computer knowledge and experience, and by the training provided to (or acquired by) employees? The surprising answer is "not much."

The 1988 data from over 3,000 employees in 46 leading-edge U.S. cities, which is used to predict the "common state" of computerization today and in the near future, indicate that training and software and user characteristics are, though, not equal in influence on computer use. Specifically, software characteristics are not major determinants of frequency of computer use. Rather, this study suggests that the computer literacy of employees is a greater factor influencing the frequency of performing all tasks except the most common, search. The number of years of computer experience that employees have also has a consistent but weaker effect on computer use for most tasks except word processing and is not statistically relevant for managers.

What these findings point to is the important role that training can play in getting employees to take advantage of the computer to help them do their jobs. Whereas software characteristics, training, and user computer background are factors conceptually independent of each other, they can also be related in that they can have a mitigating influence on each other (Kaplan & Duchon, 1988). For example, user friendliness of software can compensate for lack of user computer experience or literacy. And these latter user characteristics can compensate for lack of user friendliness or technical friendliness. At the same time, training can compensate for software unfriendliness and limited user computer background. Obviously, the industry has certain products that an organization has purchased, and an organization also has a set group of employees already on the job. The point is that if an organization has to use the computer to do a task, it does, no matter how bad the software (Gasser, 1988). Software investments are usually too large to change readilv, and systems take a long time to change. Moreover, change is fraught with problems. Thus, changes in software and/or employees are less viable options than changes in training programs. In sum, training not only can compensate for deficiencies in the user and technical friendliness of computer systems but also can compensate for the lack of computer literacy and experience of users with the least cost and trauma to the organization. The data indicate that formal training of any amount, even 1 hour, results in employees' turning to the computer more often to perform almost every kind of computer task. Furthermore, a 1989 study of 340 city managers and mayors confirms our findings from a different perspective. The responding local government officials ranked computer literacy 5th out of 30 perceived information, training, and assistance needs (Slack, 1990).

Therefore, organizations wishing to encourage computer use should promote formal training by sponsoring in-house seminars and training sessions, by promoting weekend or regular academic calendar courses in computing at nearby colleges and universities, and by sending employees to professional conferences with panels or workshops in computing.

In addition, training programs need to be sensitive to variation in impediments to computer use by different types of employees. For example, managers need training sessions that make software programs seem more user friendly, as do employees who just search, enter, or update files. In contrast, employees who process text need training to address technical glitches in the programs.

Who provides the majority of initial computing training, whether employee, co-worker, or computing professional, appears to be a complicated issue relating to use. In other words, sometimes self-training is fine as the initial major source of training for using the computer in the job. But once an employee has some experience with using the computer in his or her job, then formal training can likely expand use because specific questions, perhaps unique to an individual or too sophisticated to deal with at first, can be addressed. Such continuation or ongoing training programs might seek employee suggestions for topics to address (Carnevale & Sharp, 1993).

Ongoing training can, then, be very important but, unfortunately, has not been stressed enough in most cities. For instance, there usually is a burst of training programs when a new computer system is installed. Systems are rarely replaced for several years. Between installation of the system and replacement of that system training falls off. Hence, new employees and transferred or promoted employees do not always have the opportunity to take the same level of intensive training that was offered in the first year or two after changeover.

The result is a workforce less able to avail itself of the capabilities of the computer system. The type of ongoing computer training that an employee needs can be evaluated during performance appraisals. Applying the lessons of this study to such appraisals may improve the perceived fairness of the appraisal process because such a directive will develop plans for needed employee training (Daley, 1987).

Finally and important to note, the foregoing analysis has found that software, training, and employee characteristics significantly affect computer use, but the relationships are not strong. In fact, the r squares from regression equations ranged from 1% to 12%, which means less than an eighth of all computer use can be explained by the six factors studied. These findings are surprising given earlier literature cited in the development of this study's topic. Why the differences? One reason is methodological. Prior studies were case studies or industry-specific studies in which some variables can be very important to computer use but will look atypically important when contrasted with 3000 employees in 46 cities. The latter data cut across organization and employee idiosyncrasies and thus can very well mute the influence of variables that are key in one setting but not consistently so across settings. Still, small changes in computer use can result in greater efficiency and effectiveness. Even a change as little as 1% can be worth it, especially for large cities.

It is also important to emphasize that this study supported our hypothesis that software, training, and user characteristics have differential effects depending upon the type of task performed and type of employee, broadly defined, performing the task.

Clearly, other factors besides those addressed in this article influence computer use. An obvious candidate is the demands of the job. For example, must an employee enter every traffic ticket into the computer or does the employee have discretion to select only some cases to search computer files for information as managers do? The data analyzed are from leading-edge cities, so tasks are highly automated. Degree of automation is not correlated with use in such cities because automation does not really vary. Other influences found have been the political climate (Kraemer et al., 1981), top management commitment (Weill & Olson, 1989), and standard operating procedures (Sanders & Courtney, 1985).

The bottom line, though, is that training can be the key to linking people and machine. Of utmost importance, when employees have to use computer systems, we have found that they will use them regardless of the quality of the software and their own lack of experience or knowledge. Thus training is the management policy for obtaining greater computing payoffs because training is not just simply a link between people and machine: it can be the versatile substitute for deficiencies of people and machine.

Unfortunately, our recommendations on the versatility of training are not wholly surprising and yet are not being addressed enough in either the public or the private sector. "Research has already shown that learning in school and on the job is by far the most important factor accounting for America's economic growth and productivity in this century" (Carnevale, Gainer, & Villet, 1990). The American Study for Training and Development recommends employers spend 2% of their payroll on formal training and 4% in the long term. Large corporations are spending less than 2% (Carnevale & Gainer, 1988). There is a paucity of comparable data in the public sector, but the investment is far less than in large private corporations. Hence, government's investment in training cannot be underscored enough. "Training is a critical foundation for all employees" (Wagenheim & Pevrink, 1991), and should be a priority, especially in tough times (Barnes, 1992).

Notes

Alana Northrop is a professor of political science at California State University, Fullerton. Her most recent research has been in the areas of information systems, organizational assessment, and the responsiveness and effectiveness of the federal bureaucracy. She can be reached at: Department of Political Science, California State University–Fullerton; Fullerton, CA 92637; 310-438-2606; FAX 310-987-3776.

Kenneth L. Kraemer is a professor in the Graduate School of Management and the Department of Information and Computer Science at the University of California, Irvine. He is also director of the Center for Research on Information Technology and Organizations. His most recent books are *Datawars* (1987), *Wired Cities* (1987), and *Managing Information Systems* (Jossey-Bass, 1989). He can be reached at: Center for Research on Information Technology and Organizations, University of California–Irvine; Irvine, CA 92717; 714-856-5246; FAX 714-856-8091; E-mail KKRAEMER@UCI.EDU.

Debora E. Dunkle is a research specialist at the Center of Research on Information Technology and Organizations at the University of California, Irvine. She is co-author of *Managing Information Systems* (Jossey-Bass, 1989) and can be reached at: Center for Research on Information Technology and Organizations, University of California, Irvine, Irvine, CA 92717; 714-856-7098; FAX 714-856-8091; E-mail DEDUNKLE@UCI.EDU.

John Leslie King is a professor in the Department of Information and Computer Science and the Graduate School of Management at the University of California, Irvine. He is a co-author of *Dynamics of Computing* (1985), *Datawars* (1987), and *Managing Information Systems* (Jossey-Bass, 1989) and can be reached at: Information and Computer Science, University of California, Irvine, Irvine, CA 92717; 714-856-6388; FAX 714-856-4056; E-mail JLKING@UCI.EDU.

1. URBIS stands for Urban Information Systems, which is a 25-year panel study of the management, use, and impact of computerization in U.S. local governments. Data for the first panel were collected in 1975 and 1976, for the second panel in 1985 and 1987–88. Data collection for the third panel will begin in 1995.

2. The cities studied in 1988 included Albany, NY; Atlanta, GA; Austin, TX; Baltimore, MD; Bellevue, WA; Bloomington, MI; Boulder, CO; Brockton, MA; Burbank, CA; Charlotte, NC; Chesapeake, VA; Cleveland, OH; Costa Mesa, CA; Evansville, IN; Ft. Lauderdale, FL; Grand Rapids, MI; Hampton, VA; Kansas City, MO; Lancaster, PA; Las Vegas, NV; Lincoln, NB; Long Beach, CA; Miami Beach, FL; Milwaukee, WI; New Orleans, LA; New Rochelle, NY; Newton, MA; Oshkosh, WI; Paterson, NJ; Philadelphia, PA; Phoenix, AZ; Portsmouth, VA; Provo, UT; Quincy, MA; Richardson, TX; Riverside, CA; Sacramento, CA; San Francisco, CA; San Jose, CA; Seattle, WA; Spokane, WA; St. Louis, MO; Stockton, CA; Tampa, FL; Warren, MI; and Warren, OH.

3. There is no exact response rate because questionnaires were allotted to cities first by their size (160 for cities 100,000 or larger in population and 130 for smaller cities) and second by the number of department heads and relevant computer users in certain positions, which varied by city. Thus response rates based on sheer allocation by size

is a conservative 71%. Given that some cities had, for example, fewer department managers and division heads, their relevant sample size would actually be less than 160 or 130, whichever was the initial questionnaire allotment. The sampling frame can also be computed only roughly. The total number of computer users in each city is minimally equal to the number of computer terminals and microcomputers given that several workers can share a terminal or microcomputer. The sample size represents approximately one quarter the total number of terminals and microcomputers. Put another way, the sample size has a confidence interval of +1.5% with a confidence level of 99%. These sampling errors are strictly relevant when the theoretical population is all computer users rather than cities.

4. We debated this decision. Variation in use clearly ranges from none to constant daily use. But if an employee never uses a computer, then degree of variation in use is not the issue, instead it is the necessity of computer use at all. In the end, the data told us that the debate was more important theoretically than in fact because the trend of correlations was the same whether one included the full sample or just employees who at least once a year performed one computer task.

5. The analysis of police computer use generated an article-length discussion on its own. The findings parallel the ones for nonpolice users in regard to hypotheses 1, 3, and 5 for all police users and hypothesis 4 for detectives only.

6. A classic methodological issue is that the order of questions can set up response patterns or biases. It should be noted that the order of the questions in the questionnaire began with the degree of initial professional training followed by hours of formal training and then by a question on number of hours spent on self-training, in order to clearly distinguish the measures sought. In this way, the questionnaire attempted to prompt the respondent to recall the learning process and to distinguish between formal and self-training both in the initial learning process as well as in the subsequent learning process.

References

Barnes, R. J. 1992. Training in tough times should be a priority. *PA Times* 15:8.

Carnevale, A. P., & Gainer, L. J. 1988. *The learning experience*. Alexandria, VA: American Society for Training and Development.

Carnevale, A. P., Gainer, L. J., & Villet, J. 1990. *Training in America*. San Francisco: Jossey-Bass Publishers.

Carnevale, D. G., & Sharp, B. S. 1993. The old employee suggestion box. *Review of Public Personnel Administration* 13 (Spring): 82–92.

Cheney, P. H., Mann, R. I., & Amoroso, D. L. 1986. Organizational factors affecting the success of end-user computing. *Journal of MIS* 3 (Summer): 65–80.

Daley, D. 1987. Performance appraisal and the creation of training and development expectations: A weak link in mbo-based appraisal systems. *Review of Public Personnel Administration* 8 (Fall): 1–10.

Danziger, J. N., & Kraemer, K. L. 1985. Computerized data-based systems and productivity among professional workers: The case of detectives. *Public Administration Review* 45 (January/February): 196–209.

———. 1991. Survey research and multiple operationalism: the URBIS project methodology. In *The information systems research challenge: Survey research methods*, vol. 3, ed. Kenneth L. Kraemer, 351–72. Cambridge: Harvard Business School Press.

Davis, F. D. 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly* 13:319–39.

Davis, F. D., Bagozzi, R. P., & Warshaw, P. R. 1989. User acceptance of computer technology: A comparison of two theoretical models. *Management Science* 35:982–1003.

DeSanctis, G., & Courtney, T. F. 1983. Toward friendly user MIS implementation. *Communications of the ACM* 26:732–38.

Dosi, G., Freeman, C., Nelson, R., Silverberg, G., & Socte, L., eds. 1988. *Technical change and economic theory*. New York: Pinter Publishers.

Fox, J. 1991. *Regression diagnostics*. Newbury Park, CA: Sage University Papers.
Gasser, L. 1988. The integration of computing in routine work. *ACM Transactions on Office Information Systems* 4 (July): 205–25.
Kaplan, B., & Duchon, D. 1988. Combining qualitative and quantitative methods in information systems research: a case study. *MIS Quarterly* 12:571–86.
Kling, R., & Jewett, T. 1991. The dynamics of computer use in a social science research team: A case study of infrastructure strategies and skill. *Social Computing Review* 9:246–75.
Kraemer, K. L., Dutton, W. H., & Northrop, A. 1981. *The management of information systems*. New York: Columbia University Press.
Kraemer, K. L., & Northrop, A. 1989. Curriculum recommendations for public management education in computing: An update. *Public Administration Review* 49:947–53.
Leonard-Barton, D., & Deschamps, I. 1988. Managerial influence in the implementation of new technology. *Management Science* 34:1252–65.
Lucas, Jr., H. E. 1989. *Managing information services*. New York: Macmillan Publishing.
Mohan, L., Holstein, W. K., & Adams, R. B. 1990. EIS: It can work in the public sector. *MIS Quarterly* 14:435–48.
Newcomer, K., & Caudle, S. 1991. Evaluating public sector information systems: More than meets the eye. *Public Administration Review* 51:377–84.
Northrop, A., Dutton, W. H., and Kraemer, K. L. 1982. The management of computer applications in local government. *Public Administration Review* 42:234–43.
Northrop, A., Kraemer, K. L., Dunkle, D., & King, J. L. 1990. Computer payoffs: Lessons over time. *Public Administration Review* 50:505–14.
Rivard, S., & Huff, S. L. 1988. Factors of success for end-user computing. *Communications of the ACM* 31:552–61.
Rosenberg, N. 1982. *Inside the black box: Technology and economics*. Cambridge: Cambridge University Press.
Sanders, L. G., & Courtney, J. F. 1985. A field study of organizational factors influencing DSS success. *MIS Quarterly* 9:77–89.
Sein, M. K., Bostrom, R. P., & Olfman, L. 1987. Training end users to compute: Cognitive, motivational social issues. *Infor* 25 (August): 236–55.
Shangraw, R. F. 1986. How public managers use information: An experiment examining choices of computer and printed information. *Public Administration Review* 46 (Special Issue): 506–15.
Slack, J. D. 1990. Information, training, and assistance needs of municipal governments. *Public Administration Review* 50:450–57.
Srinivasar, A. 1985. Alternative measures of system effectiveness: Associations and implications. *MIS Quarterly* 9:243–53.
Wagenheim, G. D., & Pevrink, J. H. 1991. Customer service in public administration. *Public Administration Review* 51:263–70.
Weill, P., & Olson, M. H. 1989. Managing investment in information technology: Mini-case examples and implications. *MIS Quarterly* 13:3–17.
Weiss, A. R., & Birnbaum, P. H. 1989. Technological infrastructure and the implementation of technological strategies. *Management Science* 35:1014–26.
Zmud, R. W., & Lind, M. R. 1985. Linking mechanisms supporting end-user computing. Proceedings of the 12th Annual Conference of ACM SIGCPR/SIGBDP, 74–80.